25885 DUP.

NRC 25885

UNIVERSITÉ DU QUÉBEC À HULL Actes du 4e colloque sur la formation professionnelle

Titre: Micro NATAL et son système-auteur

Conférencier(s): Benoit Farley, ing. M.Sc.A.
Conseil national de recherches

NATAL

NATAL est un langage de programmation à haut niveau, structuré et modulaire, spécialement conçu pour l'enseignement assisté par ordinateur. Le nom de ce langage, **NATAL**, est un acronyme dérivé de **"NATional Author Language"**, ou langage d'auteur national, et **micro-NATAL** correspond à la version de ce langage pour micro-ordinateur.

Un programme écrit en **NATAL** est construit à partir de trois éléments principaux: les PROCÉDURES, les FONCTIONS et les UNITÉS, auxquels viennent se greffer les FICHIERS. Les procédures et les fonctions sont des routines modulaires dont la forme et la syntaxe se retrouvent pour l'essentiel dans plusieurs autres langages structurés à haut niveau. Elles sont utilisées principalement pour la gestion du cheminement du cours et pour le calcul et l'établissement de variables diverses. L'UNITÉ est l'élément qui caractérise vraiment NATAL en tant que langage d'auteur. C'est par cette routine spéciale que passent toutes les communications et les transactions instructionnelles entre le cours (l'ordinateur) et l'étudiant.

UNITÉ - SOUS-LANGAGE DE REPRÉSENTATION

L'UNITÉ NATAL est formée de cinq parties:

1. l'affichage d'information sous formes écrite, graphique, parlée et picturale
2. l'obtention d'une réponse sur clavier, sur écran sensible au toucher et sur tablette numérique

3. l'édition de la réponse obtenue pour enlever l'information superflue
4. la classification de la réponse obtenue par comparaison avec des réponsescibles prédéfinies en catégories
5. l'affichage d'information additionnelle selon la réponse obtenue, et la fin de la transaction ou le retour à l'obtention d'une nouvelle réponse

Une UNITÉ peut ne contenir qu'une simple opération d'affichage sur écran, comme elle peut être construite pour effectuer une transaction très complexe avec plusieurs catégories de réponse, plusieurs essais pour chaque catégorie et un affichage spécifique pour chacun, intégrant des commandes pour faire fonctionner un projecteur de diapositive ou un lecteur de cassette à accès direct, un vidéodisque, etc... La figure 1 montre un exemple d'une UNITÉ NATAL.

```
*U04204:  UNITÉ;

AFFICHER &N &B(3,7,4,27) Laquelle des illustrations montre la bonne façon d'effectuer les
         coups entre les omoplates?
         &B &G(DEF CLOTURE, R425A PI(-20,50), R425B PI(40,6) )
         &C(9,4) A
         &C(9,22) B
         &C(18,5) Réponse:
         &G(DEF CLOTURE, DEF FENETRE(0,0,256,200), DEF COULEUR(5),
            DÉPLACER(81,19), RPOLY( (%9,0,-9%),(%0,12,0%) ) )
         &R &;

RÉPONSE POS=(18,14), NCAR=2;
ÉDITER PASMAJ;

EXACT  CC('B');
INEXACT CC('A);

REINF EXACT &E(18,17,1,16) En effet! &R &;

REINF INEXACT &E(18,17,1,16) &HC(5) NON. &HC(8) Les coups ne sont
         &C(19,5) pas donnés entre les omoplates. &R &;

RESSAYER NONREC &E(18,17,1,16)
                &HC(8) Répondre &HC A &HC(8) ou &HC B
                &W(2)
                &E(18,14,1,2)
                &G(DEF CLOTURE, DEF FENETRE(0,0,256,200), DEF COULEUR(5),
                DÉPLACER(90,19), RLIGNE(0,12) )
                &R &;

FIN;
```

Figure 1. Exemple d'une UNITÉ NATAL

L'affichage de l'information et la commande de périphériques est assurée par le sous-langage de représentation (SLR) de NATAL, constitué de commandes simples à haut niveau couvrant une large gamme d'opérations. Le tableau 1 contient la liste des commandes du SLR avec une brève description de chacune. Le tableau 2 contient la liste des commandes graphiques du langage NATAL.

&A	Début du mode d'affichage non formaté du texte
&B	Définition d'une boîte pour la mise en forme du texte
&C	Déplacement direct du curseur de texte à l'écran
&D	Commande des dispositifs: projecteur de diapositives, audio, spéciaux
&E	Effacement d'une surface rectangulaire de l'écran
&F	Sélection d'un jeu de caratères, avec chargement au terminal si nécessaire
&G	Début du mode d'affichage graphique
&HB	Définition de la luminosité du texte
&HC	Définition de la couleur du texte
&HD	Définition de la fréquence de clignotement du texte
&HG	Définition de la couleur d'effacement
&I	Définition des marges horizontales
&J	Justification du texte entre les marges horizontales
&K	Détermination du nombre de lignes disponibles dans la boîte
&L	Déplacement du curseur de texte vers le bas et retour à la marge gauche
&M	Centrage du texte entre les marges horizontales
&N	Début d'une nouvelle page avec attente avant effacement de la page
&O	Surimpression des caractères
&P	Début d'un nouveau paragraphe
&Q	Modification du caractère de commande du SLR (par défaut: &)
&R	Réinitialisation des paramètres d'affichage
&S	Définition de l'espacement entre les lignes de texte formaté
&T	Définition de la position du dernier caractère d'un mot
&U	Soulignement du texte
&V	Insertion de la valeur d'une variable dans la séquence de texte
&W	Définition d'une pause, en secondes
&X	Transmission de codes ascii au terminal
&YC	Détermination de la position courante du curseur à l'écran
&YF	Chargement d'un jeu de caractères au terminal
&Z	Fin du monde d'affichage non formaté et début de la mise en forme du texte
&-	Identification de la position de coupure d'un mot en fin de ligne
&;	Fin de l'instruction d'affichage dans une UNITÉ

TABLE 1. Sous-langage de représentation NATAL

ARC	Génération d'un arc de cercle
CERCLE	Génération d'un cercle
POLY	Génération d'un polygone avec coordonnées absolues
RPOLY	Génération d'un polygone avec coordonnées relatives
LIGNE	Génération d'une ligne avec coordonnées absolues
RLIGNE	Génération d'une ligne avec coordonnées relatives
MARQUE	Génération d'un symbole avec coordonnées absolues
RMARQUE	Génération d'un symbole avec coordonnées relatives
DÉPLACER	Déplacement du curseur graphique avec coordonnées absolues
RDÉPLACER	Déplacement du curseur graphique avec coordonnées relatives
CHAR	Génération de texte en mode graphique

DEF_REMPL	Définition du type de remplissage (hachurage)
DEF_CLIGNE	Définition de la fréquence de clignotement des éléments graphiques
DEF_COULEUR	Définition de la couleur des éléments graphiques
DEF_LIGNE	Définition du type de ligne
DEF_MARQUE	Définition du type de symbole
DEF_FENETRE	Définition du système de coordonnées graphiques
DEF_CLOTURE	Définition de la surface rectangulaire de l'écran associée au système de coordonnées

TABLE 2. Commandes graphiques du langage NATAL

PORTABILITÉ

Le système NATAL comporte un interface logiciel entre les commandes à haut niveau du sous-langage de représentation et le terminal utilisé: il s'agit du module de description du terminal. Celui-ci traduit les commandes du programme NATAL en une séquence de codes reconnues par le terminal pour effectuer les opérations désirées. Ainsi, pour autant que le système NATAL possède un module de description pour chacun des dispositifs d'une station de travail, l'auteur n'a pas à se soucier de connaître les séquences de contrôle d'un dispositif particulier et de préparer un cours distinct pour tel ou tel type et/ou marque de dispositif. Il n'a qu'à se préoccuper des caractéristiques du terminal. Un cours écrit en NATAL pour un certain type de terminal peut être diffusé sur tout autre terminal possédant les mêmes caractéristiques, quelle qu'en soit la marque et le protocole de communication. La portabilité des cours d'un terminal à un autre est ainsi assurée.

Le système NATAL, tel que développé aux laboratoires du Conseil national de recherches du Canada, comprend deux sous-systèmes, l'un pour la compilation des programmes NATAL en un langage intermédiaire et l'assemblage des programmes, l'autre pour l'interprétation des

instructions du langage intermédiare. Ces deux sous-systèmes sont écrits en langage "C", que l'on retrouve avec un haut degré de standardisation sur de nombreux micro-ordinateurs. De plus les parties de ces sous-systèmes qui dépendent directement du micro-ordinateur et du matériel sont clairement identifiées, ce qui réduit au minimum les efforts nécessaires au transport de NATAL d'une machine à une autre. Un cours écrit en NATAL peut ainsi être diffusé, sans aucune modification des programmes sources, sur tout micro-ordinateur supportant NATAL. La portabilité des cours est ainsi assurée.

SYSTEME-AUTEUR

Malgré la simplicité d'apprentissage que l'on reconnaît généralement au langage d'auteur NATAL (1), il n'en demeure pas moins un langage de programmation qui demande un minimum d'aptitude et d'effort que l'on ne saurait être en droit d'exiger de la part de tous les utilisateurs potentiels de l'enseignement assisté par ordinateur avec NATAL. Pour amener les auteurs à l'EAO et aussi pour augmenter la productivité dans le développement de cours, un système-auteur simple, flexible mais puissant, procurant divers niveaux de complexité d'opération pour répondre aux divers niveaux de complexité des modules d'un cours, doit être utilisé. (2)

Un système-auteur fondé sur le langage d'auteur NATAL est en cours de développement aux

laboratoires du Conseil national de recherches du Canada. Sa structure est représentée à la figure 2. Il comprend essentiellement 5 outils, appelés éditeurs:

-éditeur graphique

-éditeur de texte

-éditeur de transaction

-éditeur de contrôle

-éditeur de programme

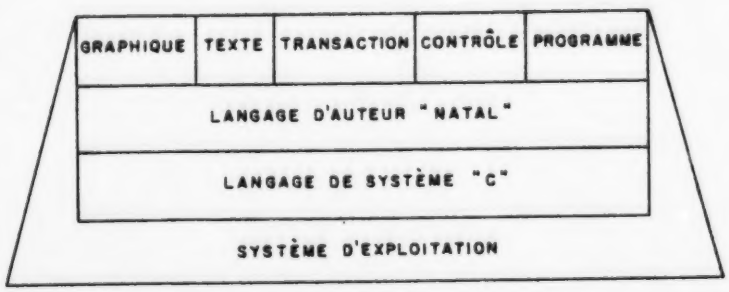

Figure 2. Structure du système-auteur NATAL

ÉDITEUR GRAPHIQUE

L'éditeur graphique est le premier des outils du système-auteur NATAL à avoir été développé. Il sert à la préparation de pages graphiques et possède de plus certaines fonctions pour manipuler le texte. Les nombreuses commandes d'édition sont placées dans un menu installé sur une tablette numérique. Les éléments graphiques peuvent être crées soit à partir d'un dessin existant que l'on fixe sur la tablette et que l'on trace, soit directement à l'écran. Les éléments graphiques de base sont: l'arc de cercle, le cercle, le rectangle, le polygone, la ligne, le marqueur et le texte. Les attributs associés aux éléments graphiques sont: la couleur (16 parmi le nombre offert par le terminal utilisé), le type de ligne (pleine, pointillée, hachurée, hachurée-pointillée), le clignotement, le type de marqueur (. 0 x +), auxquels viennent s'ajouter pour les éléments fermés (cercle, arc, polygone, rectangle) les attributs de remplissage: le type (vide, horizontal, vertical, croisé, plein), la couleur, le type de ligne et le clignotement. Les attributs associés au texte sont la couleur, le clignotement et le jeu de caractères.

L'éditeur graphique possède un certain nombre de commandes des modification des éléments graphiques: le déplacement d'un endroit à un autre, la modification des attributs, la copie à un autre endroit, la réflexion par rapport à un plan, la rotation d'un nombre donné de degrés, le changement de dimensions dans les directions horizontale et verticale, le réarrangement de l'ordre d'affichage des éléments, l'enlèvement d'éléments. On peut aussi modifier un élément de texte par la modification de ses attributs, par le déplacement à une autre position, par l'insertion de caractères, par la division d'une ligne en deux lignes successives, par la juxtaposition de deux lignes successives, par l'enlèvement de caractères. Un nombre indéfini d'éléments peuvent être regroupés pour former des objets; les commandes de modification des

éléments énumérées plus haut s'appliquent aussi aux objets. Pour identifier un élément ou un objet que l'on veut modifier, on le pointe avec le curseur graphique directement à l'écran.

D'autres commandes de support sont incorporées à l'éditeur graphique: le réaffichage de la page, l'effacement de l'écran et l'affichage sélectif d'objets, la sélection et l'affichage d'une grille de texte et d'une grille graphique, l'affichage pour référence graphique de pages graphiques et de texte déjà préparées, la copie d'éléments et d'objets distincts appartenant à d'autres pages, la création d'un objet à partir des éléments existants, le commencement d'un nouvel objet, la modification de la table de couleurs.

Les deux dernières lignes du bas de l'écran sont réservées pour l'affichage des paramètres d'édition en vigueur à tout moment pendant l'édition d'une page.

ÉDITEUR DE TEXTE

Bien qu'il soit possible avec l'éditeur graphique de générer du texte, un éditeur de texte distinct a été proposé pour simplifier la tâche de l'auteur dans les cas où il n'est prévu que du texte et/ou des graphiques prédéfinis créés à l'aide de l'éditeur graphique. Contrairement à ce dernier, toutes les commandes sont activées à partir du clavier.

L'éditeur de texte proposé possède d'abord les caractéristiques que l'on rencontre normalement dans un éditeur de texte standard. On lui a aussi ajouté plusieurs autres caractéristiques en relation avec la nature "videotex" des pages à préparer. L'information écrite est entrée sous forme de "blocs" de texte, un bloc étant défini comme l'ensemble des caractères tapés entre le moment où la commande de commencement du bloc est donnée et le moment où la commande de fin du bloc est donnée. De plus, pour chaque bloc, une marge gauche est déterminée par

l'auteur. Les attributs associés au texte sont la couleur, le clignotement, le jeu de caractères, le soulignement et la surimpression. L'auteur peut les définir avant de commencer un bloc, ou il peut aussi les modifier pendant l'édition d'un bloc. Une autre série de commandes sont offertes pour modifier, pendant ou après l'édition d'un bloc, le texte déjà entré. L'auteur peut ainsi diviser une ligne en deux lignes successives ou coller deux lignes successives bout à bout, insérer du texte devant un caractère ou ajouter du texte à la fin d'une ligne, enlever des caractères, modifier les attributs d'une suite de caractères, définir une limite gauche et une limite droite pour ensuite centrer ou justifier une ou plusieurs lignes, ou reformater un bloc après modification. Il est aussi possible de déplacer tout un bloc à un autre endroit à l'écran, de modifier sa position dans l'ordre d'affichage des divers éléments composant la page, de substituer un mot ou une expression à une ou toutes les apparitions d'un autre mot ou d'une autre expression dans la page.

Une page graphique peut être liée à une page d texte. L'auteur n'a qu'à identifier la page graphique et à définir à l'écran la surface rectangulaire à volonté les dimensions ainsi que la position de cette surface. Une liaison peut aussi être établie avec une autre page de texte. Une copie d'une autre page de texte peut être ajoutée à la page en cours d'édition. Les liaisons et les copies peuvent être établies avec n'importe quelle page créée à l'aide des éditeurs graphique et de texte, quel que soit le cours pour lequel elles ont été préparées originalement.

Un autre élément de l'éditeur de texte est la boîte rectangulaire, dont l'auteur peut faire usage pour renforcer l'affichage d'un bloc de texte ou d'un mot. La boîte peut être de deux types: cadre ou surface solide. L'auteur peut en choisir le type, les dimensions, la couleur du contour, la couleur de l'intérieur d'une boîte solide et la position. Ces attributs peuvent être modifiés une fois la boîte créée.

Enfin l'auteur a la possibilité d'identifier un mot ou une expression pour les inclure dans un glossaire. Après identification de la cible, l'auteur tape au clavier l'information pertinente s'y rattachant et qui sera présentée à l'étudiant lorsqu'il demandera à voir la définition de ce terme ou de cette expression pendant qu'il suit un cours.

ÉDITEUR DE TRANSACTION

L'éditeur de transaction sert à la préparation d'une transaction instructionnelle entre l'ordinateur et l'étudiant. Celle-ci étant basée sur l'UNITÉ NATAL, on y retrouve sa structure générale et ses principales caractéristiques.

L'auteur définit d'abord l'énoncé de la question, soit à l'aide de l'éditeur de texte, soit par liaison directe avec des pages de texte ou graphiques déjà préparées. Il peut ensuite déterminer les divers paramètres relatifs à l'obtention de la réponse: type de réponse (clavier, pointeur, etc...), temps alloué à l'étudiant pour fournir la réponse. Pour les réponses écrites, il peut spécifier le nombre de caractères maximum, l'emplacement à l'écran où la réponse sera acceptée, la couleur, le jeu de caractères, la présence ou l'absence d'un retour de chariot pour terminer la réponse, une boîte et ses attributs pour contenir la réponse. Pour les réponses sur écran sensible au toucher et sur tablette numérique, il peut spécifier le dispositif (écran du terminal, écran de projection de diapositive, identificateur de la tablette numérique) et le nombre de points ou de réponses graphiques.

L'étape suivante consiste à fournir à l'éditeur de transaction, par catégorie, les diverses réponses-cibles auxquelles sera comparée la réponse de l'étudiant. Une catégorie peut contenir plus d'une réponses-cibles. Pour les réponses écrites, la comparaison peut être faite caractère par caractère, par la recherche d'un mot-clé, sur une base numérique ou par

évaluation d'une expression algébrique. L'auteur peut spécifier, par l'utilisation de caractères spéciaux tout-usage, la tolérance de son choix sur l'orthographe de la réponse de l'étudiant. Pour les réponses graphiques et par pointeur, l'auteur doit fournir les limites de la surface rectangulaire où la réponse doit être obtenue.

Pour chacune des catégories de réponse identifiées, ainsi que pour le cas d'un dépassement du temps alloué, l'auteur peut définir un message de renforcement qui sera présenté à l'écran après obtention d'une réponse, et indiquer quelle action devra être entreprise suite à ce message: terminer la transaction ou obtenir une nouvelle réponse. Dans ce dernier cas, l'auteur peut spécifier le nombre de fois qu'une nouvelle réponse pourra être demandée et un message spécifique pour chacune. Les messages de renforcement sont aussi introduits à l'aide de l'éditeur de texte ou par liaison directe avec une page graphique ou de texte déjà préparée.

Lors de la préparation d'une transaction, l'auteu a le choix entre utiliser un modèle prédéfini (VRAI ou FAUX, choix multiple, liste de paires correspondantes) où il n'a qu'à fournir les données pertinentes, et utiliser le modèle général qui procure une flexibilité complète quant à la mise en forme des données, le traitement de la réponse et la complexité de la stratégie transactionnelle.

La préparation, ou CRÉATION, est l'une des quatre fonctions de base de l'éditeur de transaction. Les trois autres dont: LISTE, REVUE et VÉRIFICATION. La CRÉATION d'une transaction, grâce à l'utilisation de l'éditeur de texte et des liaisons avec d'autres pages, est effectuée de telle manière que l'auteur, pendant le développement du contenu, voit exactement à l'écran ce qui sera présenté à l'étudiant. Ceci élimine donc complètement la nécessité de recourir à des modèles d'écran sur papier pour préparer la mise en forme du texte à l'écran. La fonction LISTE permet d'obtenir à l'écran une liste des réponses-cibles, des messages de

renforcement qui leur sont associés et des actions à entreprendre suite aux messages. La fonction REVUE permet à l'auteur de passer en revue la totalité de la transaction pour chacune des catégories de réponse. Enfin la fonction VÉRIFICATION donne à l'auteur le moyen de vérifier la transaction avec des réponses-tests.

ÉDITEUR DE CONTROLE

L'éditeur de contrôle, pilier du système-auteur, permet à l'auteur d'assembler les divers éléments générés par les éditeurs graphique, de texte et de transaction, en plus des programmes directement codés en NATAL, selon une structure de contrôle qu'il détermine lui-même. L'enchaînement des blocs d'information (pages de texte et graphiques) ne pose pas de problème étant donné leur totale modularité. On peut ainsi ajouter ou enlever de tels blocs dans un enchaînement sans en modifier la structure. Par contre les blocs de transaction constituent des noeuds, des points de branchement potentiels. L'éditeur de contrôle reconnaît ce type de bloc et permet à l'auteur de définir la suite de l'enchaînement selon les divers résultats possibles de la transaction. Pour ce faire l'auteur a accès à un certain nombre de variables, tels la réponse finale de la transaction, le nombre d'essais pour une nouvelle réponse effectués par l'étudiant, le temps écoulé... Il peut aussi spécifier si les résultats et autres statistiques de la transaction doivent être enregistrés ou pas, le nombre de points alloués pour chaque catégorie de réponse. Lorsqu'un branchement est identifié par l'auteur, celui-ci poursuit avec l'enchaînement de chaque embranchement à tour de rôle. Les blocs de transactions ne sont pas les seules sources de branchement cependant. À tout moment pendant l'établissement du cours, l'auteur peut indiquer un branchement basé sur les performances de l'étudiant, sur le nombre de fois qu'une leçon ou qu'une partie de leçon a été vue, etc...

L'éditeur de contrôle prévoit aussi des fonctions utilitaires de contrôle que l'auteur peut définir dans un cours pour aider l'étudiant. Il s'agit d'abord de la fonction GLOSSAIRE, par laquelle l'étudiant peut demander des détails supplémentaires sur un terme ou sur une expression; après l'avoir obtenue, l'étudiant revient là où il était. Il y a ensuite la fonction AIDE qui permet à l'étudiant d'obtenir une aide quelconque sur la marche à suivre et les diverses fonctions disponibles, la fonction RAPPEL qui permet à l'étudiant de revoir une page ou un chapitre déjà vu, la fonction ARRET qui permet à l'étudiant de terminer le cours temporairement pour revenir plus tard là où il s'était arrêté.

L'éditeur de contrôle permet la construction d'un cours en leçons et chapitres et peut générer de lui-même, une fois l'enchaînement d'une leçon terminé, la table des matières avec ou sans numéros de page pour cette leçon. L'auteur peut spécifier si oui ou non les tables des matières des leçons seront présentées au début des leçons. La table des matières peut être présentée lors de l'utilisation de la fonction RAPPEL. La table des matières du cours peut aussi être générée lorsque le cheminement du cours est complètement terminé.

ÉDITEUR DE PROGRAMME

Bien qu'un programme NATAL puisse être écrit avec l'éditeur de texte standard du système d'exploitation, l'éditeur de programme facilite grandement la tâche de l'auteur pendant la programmation en vérifiant à tout instant les erreurs possibles de syntaxe et d'orthographe des mots réservés du langage, tout en lui offrant les possibilités d'un éditeur de texte standard. De plus l'éditeur de programme permet d'exécuter le module programmé et de voir ainsi le résultat comme le verra l'étudiant. L'utilité d'un tel éditeur est évidente si l'on pense par exemple aux boucles et aux blocs SI-ALORS-SINON imbriqués à plusieurs niveaux, où il est courant, même pour un programmeur expérimenté, d'oublier de fermer une boucle ou un bloc conditionnel. De telles erreurs et oublis modifient parfois profondément la logique d'un

programme et sont souvent difficiles à déterminer, sans compter le temps perdu à corriger et à recompiler.

La possibilité dans un système-auteur d'utiliser des modules programmés en augmente considérablement la flexibilité et la puissance. Bien que la plupart des auteurs demeurent généralement au niveau des outils spécialisés (éditeurs), certains autres ont vite tendance à pousser le système-auteur au-delà de ses limites dès qu'ils sont devenus familiers avec ses possibilités. Il est donc important qu'un système-auteur puisse répondre aux exigences et aptitudes diverses des auteurs aussi bien qu'aux exigences définies par les applications projetées.

Dans ce sens l'utilisation de modules programmés en NATAL permet l'appel de programmes écrits en langage "C", et de là d'accéder au système d'exploitation du micro-ordinateur, ce qui multiplie considérablement les possibilités du système-auteur. Elle permet aussi la préparation de modules qui, de par la complexité de leur logique, ne sauraient être préparés à l'aide des éditeurs spécialisés du système-auteur.

AUTOGÉNÉRATION DES INSTRUCTIONS NATAL

Sauf dans le cas de l'éditeur de programme, dont la sortie consiste déjà d'instructions NATAL, chaque éditeur possède une fonction pour autogénérer les instructions NATAL à partir des données enregistrées dans les fichiers de données du système-auteur. Les procédures, fonctions et unités ainsi générées peuvent alors être compilées (sans erreurs) et assemblées pour former un cours NATAL qui sera diffusé à l'extérieur du système-auteur, et aussi pour être exécutées à l'intérieur du système-auteur pendant l'édition des divers modules.

CONCLUSION

On retrouvera dans le système-auteur NATAL proposé plusieurs des caractéristiques d'autres systèmes-auteurs, car elles sont évidemment fondamentales. Les différences portent surtout sur la fonctionnalité du système et sur la qualité des interactions entre celui-ci et l'auteur. De ce point de vue le langage d'auteur NATAL, sur lequel est fondé le système-auteur NATAL, lui procure à plusieurs égards une plus grande richesse, sans oublier la portabilité du système-auteur et des cours qui n'est pas le moindre des avantages des points de vue pratique et économique.

L'un des points les plus importants est que l'auteur, pendant qu'il prépare un module avec les éditeurs graphique, de texte et de transaction, voit lui-même à l'écran ce qui sera présenté à l'étudiant: l'auteur n'a pas demander l'affichage du module après avoir introduit une série de commandes, car l'affichage se fait en même temps. En ce qui concerne les autres éditeurs, de contrôle et de programme, il a la possibilité d'exécuter, sans sortir du système-auteur, les modules préparés.

RÉFÉRENCES

1. BRAHAN, J.W. and WESTROM, M.L., "NATAL-74: first results", Proc. ACDIS Conference: New Directions in Educational Computing, Dallas, Texas, 1-4 mars 1978.

2. FAIRWEATHER, P.G. and O'NEAL, A.F., "The impact of advanced authoring systems on CAI productivity", Journal of Computer-Based Instruction, Summer 1984, Vol. 11, No. 3, 90-94.